International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

# Optimizing Hyperledger dApp Efficiency: Gas Consumption in Hardhat and Truffle Frameworks

**M.Shakila*[1], A.Rama[2], P.Thilagavathi [3], M.Prakash[4], L.Anitha[5]**

**Submitted**: 03/05/2024 **Revised**: 16/06/2024 **Accepted**: 23/06/2024

**Abstract:** Blockchain technology has completely changed how decentralised applications (dApps) operate by improving security, transparency, and efficiency to previously unheard-of levels. It is imperative that the use of specialized tools result in the effective creation of the frameworks and tools used in this process. The crucial measures that are part of the computational effort and operations that go into developing the development of enhanced immutability. Ether payments, which stand for the transaction fees associated with each transaction, are necessary for the smart contract to be executed. This research explores the use of blockchain technology in voting to improve security, increase transparency, and deal with problems like manipulation and fraud. The usefulness of the Hardhat and Truffle frameworks is examined. An execution is conducted on the utility of Hardhat and Truffle frameworks for smart contract development, supporting developers in achieving cost-efficiency by embedding gas estimation and customizable gas configuration within the system. This analysis aims to provide a thorough assessment of gas consumption as compared to transactions using Ethereum Truffle, with the goal of identifying the most efficient framework for smart contract deployment. The primary objective is to compare gas usage metrics in evaluating these frameworks. The methodology entails deploying identical gas consumption measurements to assess the cost-effectiveness of implementing Blockchain-based systems. This involves evaluating gas consumption across various transaction scenarios to optimize ether usage. Consequently, data is securely stored and accessed by applications through frontend interfaces and Ethereum nodes, ensuring the immutability of systems. Collaboration between frontend and backend components is crucial in developing user-friendly and secure systems. In Future ,we present a framework for conducting optimization experiments in a distributed setting and for storing both the environmental result data and environmental studies. This framework facilitates not only the organized implementation of experiments but also the potential reuse of outcomes in subsequent endeavors.

***Keywords:*** *Blockchain, Ethereum, Hardhat ,ether, transparency, and immutability.*

## 1. Introduction

The findings highlight the superiority of one framework in achieving statistically significant outcomes in terms of gas consumption efficiency. wing to the fees users pay for resource consumption, gas consumption is an important factor to take into account when using Ethereum smart contract development frameworks like Hardhat and Truffle. External code, storage, and transaction base costs are some of the aspects that must be optimised in smart contracts to minimise user-paid expenses. Among blooming technology, the Hyperledger has emerged as robust framework in developing the dApp in the choice of implementation of the smart contract In order to encourage, mechanism seeks to match transaction fees with computational costs. The examination of this dataset is focused on providing insights for the selection of the most efficient framework for implementing online voting contracts on the blockchain with regards to gas usage. [1][2]Through a comparison of gas consumption metrics between Ethereum Hardhat and Truffle, developers are able to assess the effectiveness and cost-efficiency of each framework in the deployment and execution of smart contracts. This data empowers decision-makers to choose the optimal framework for creating gas-efficient and scalable online voting systems on the Ethereum blockchain [2][3]. The incorporation of blockchain technology into online assessment systems introduces a revolutionary method for enhancing the credibility and effectiveness of educational evaluations. optimizing gas consumption in blockchain frameworks like Hyperledger using tools such as Hardhat and Truffle is paramount for achieving environmental sustainability and cost-effectiveness. The execution of smart contracts on platforms like Ethereum and Hyperledger Fabric necessitates gas for deployment, underscoring the importance of minimizing gas expenses through code optimization methodologies. Recent studies indicate that enhancing loop control structures in smart contracts can result in notable reductions in gas consumption, with an average decline of 21% in gas costs per transaction. Furthermore, investigations into Hyperledger Fabric have revealed that enhancing energy efficiency, such as utilizing FPGA accelerators, can boost throughput by up to 10 times while maintaining consistent energy consumption levels. By

*[1] Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Saveetha University, Chennai, 602105, India*
*ORCID ID : 0009-0001-4051-5768*
*[2] Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Saveetha University, Chennai, 602105, India*
*[3] Arupadai veedu institute of Technology, Vinayaka mission's research Foundation (DU), Paiyanoor, chennai-603104, India*
*[4] Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Saveetha University, Chennai, 602105, India*
*[5] S.A Engineering College (Autonomous), Avadi , Chennai, India*
** Corresponding Author Email: shakilam1006.sse@saveetha.com*

incorporating these optimization strategies, developers can optimize the efficacy of Hyperledger dApps, thereby diminishing environmental repercussions and operational expenditures.

### 1.1. Abbreviations and Acronyms

We offer a platform for storing environmental research and environmental outcome data, as well as for carrying out optimisation experiments in a distributed context. This framework makes it easier to conduct experiments in an organised manner and to potentially reuse the results.Decentralised Applications, or dApps.

Applications that run decentralised from a central authority on a P2P network or blockchain.

Gas: A unit of measurement for the computational load on the Ethereum blockchain that goes into carrying out transactions or smart contract execution.

Ether

The Ethereum platform's native coin, which is used to cover transaction costs and computational services.

Ethereum Virtual Machine, or EVM

Ethereum's smart contract runtime environment, which enables code to function precisely as intended.

IDE stands for Integrated Development Environment, a software that gives computer programmers all the tools they need to build software. Hardhat and Truffle both provide IDE features.

Application Binary Interface, or ABI

An Ethereum network-wide contract interface standard is essential for dApp development.

JavaScript Object Notation - Remote Procedure Call, or JSON-RPC

An Ethereum network communication protocol for distant procedure calls that is represented in JSON.

Command Line Interface, or CLI

An interface that is text-based and used to communicate with operating systems and applications, as the ones offered

## 2. Related Works

Average gas cost savings per transaction around 21%. distributed digital record that is resistant to tampering and tampering, blockchain is often deployed decentralised, with no central authority.[1] Recording transactions in a shared ledger among members of a community is made possible by it. Optimize gas consumption of smart contracts, specifically loop control structures. Prototype implementation using off-the-shelf tools for Solidity smart contracts. [1]. Gas efficiency test of Hands-up-go contract using private network. Onetime Deploy method is more gas-effective when a contract contains multiple events. Ethereum's smart contract advantages for proof of existence and content. Gas-efficient event DApp named Hands Up-Go developed and tested. Development of a gas-efficient event DApp named Hands Up-Go.[2]Prevention of fraud through anonymous event holding and participation Once hashed, it cannot be decrypted back into its original form. Each block consists of two components: the block header and the block contents[13].

To compare gas consumption between Hardhat and Truffle for each operation,we can use the formula for calculating the difference in gas consumptio

$$Difference = Gas\ Consumption\ (Truffle) - Gas\ Consumption\ (Hardhat) \quad (1)$$

(1)Validate Vote

DifferenceValidate Vote=20000−18000=2000 gas (2)

(2)Tally Vote

DifferenceTally Vote=15000−14000=1000 gas (3)

(3) Update Voter Registry

DifferenceUpdate Voter Registry=25000−22000=3000 gas (4)

(4) Generate Report

$$DifferenceGenerate\ Report = 30000 - 28000 = 2000\ gas \quad (5)$$

## 3. Materials and Methods

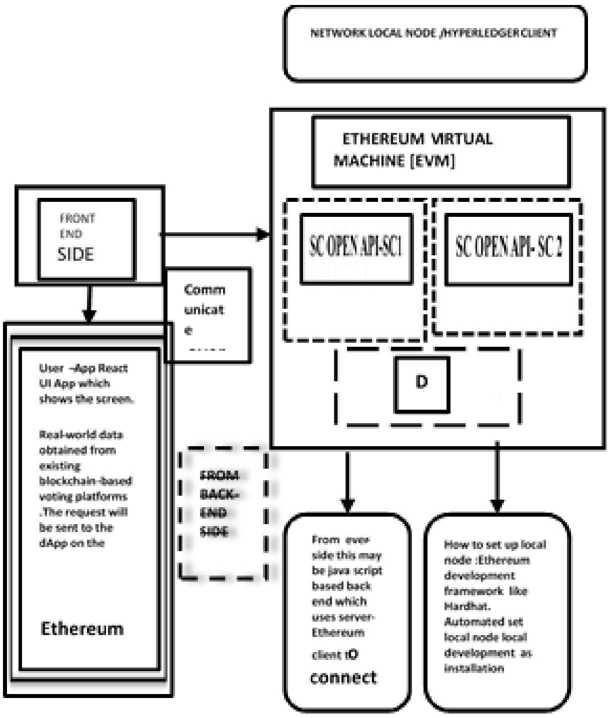


**Fig. 1.** Architecture of Blockchain Hyperledger

(i) Development Frameworks

Hardhat is recognized for its advanced debugging tools and plugin system, offering a flexible and extensible Ethereum development environment. Truffle, on the other hand, provides automated testing, scriptable deployment, and a robust developer community, making it a comprehensive development framework for Ethereum.

(ii)Blockchain Networks

The Ethereum Test Network serves as a simulated environment for deploying and testing smart contracts, facilitating gas consumption analysis. In contrast, Hyperledger Fabric, an open-source enterprise-grade blockchain framework, is utilized for decentralized application deployment, operating differently from Ethereum concerning gas usage.

(iii)Smart Contracts

Smart contracts in Solidity are standardized and intended for tasks like token transfers, data storage, and access control, ensuring efficient and reliable performance.

(iv)Development Tools

The Solidity Compiler (sol) is crucial for compiling smart contracts, while Node.js is necessary for executing JavaScript code in both Hardhat and Truffle environments. Additionally, Ganache, a personal blockchain for Ethereum development, is instrumental in deploying contracts and evaluating their performance.

## 4. The following set up considered for the set up of the Implementation

### 4.1. Construction of Smart Contract

The Ethereum test network may then be configured using Ganache to create a virtual blockchain environment for testing and deploying smart contracts.

### 4.2. Configuring the Framework

It is imperative to create standardised Solidity smart contracts with shared capabilities. To maintain consistency in measuring petrol use, these contracts should abide by the ERC20 token rules, data storage requirements, and sophisticated operations.

### 4.3. Deployment and Testing

Hardhat configuration include creating a project directory, installing necessary plugins, and configuring the Solidity compiler. Similarly, setting up the project directory, initialising Truffle, and configuring the Solidity compiler are all part of configuring Truffle.

An important first step is to deploy standardised smart contracts utilising both Hardhat and Truffle on the Ethereum test network. Token transfers, data entry, and contract calls are ways to interact with the deployed contracts.

### 4.4. Gas Consumption Comparison

Recording gas consumption at each stage of the smart contract lifecycle (compilation, deployment, interaction) is imperative. Comprehensive data collecting is made possible by utilising Hardhat and Truffle's gas reporting features.

Comparing gas consumption data from both frameworks is essential to identify patterns, efficiencies, and inefficiencies. Analyzing the data using statistical methods helps assess the significance of differences in gas consumption.

### 4.5. Implementation

Figures compiled of more than one sub-figure presented side-by-side, or stacked. If a multipart figure is made up of multiple figure types (one part is lineart, and another is grayscale or color) the figure should meet the stricter guidelines.

### 4.6. Metamask Execution for Executing Smart Contract

1) Ensure that MetaMask is connected to the Ethereum network where the smart contract is deployed in order to establish a connection.

2) Interact with the contract in Remix by utilizing a web interface or directly calling the add voter, add candidate, and map voter functions. Enter the relevant details for the candidate functions.

3) Each transaction must be confirmed and sent through MetaMask to validate the process.

4) Utilize Ether scan or an equivalent tool to oversee the transactions and verify the gas consumption

## 5. Results and Discussion

### 5.1. Calculation on Gas Consumed

In Ethereum, calculating gas usage is a standard procedure. Many Ethereum clients have integrated methods for gas estimates, such as Geth and Parity. From Table I and Table 2,A JSON-RPC method called eth_estimateGas is used to provide a gas estimate for a transaction.

Relative Gas Savings for Deployment (Hardhat vs. Truffle)

Relative Gas Savings Deployment

= 120,000−110,000/120,000 ×100%= 10,000/120,000 ×100%≈8.33%

**Table 1.** comparison of average gas consumed-blockchain based system

| Deployment | Truffle | Hardhat | Average gas cost |
|---|---|---|---|
| **Transactions 1** | 1,10,000 | 1,00,000 | 1,05,000 |
| **Transactions 2** | 1,10,000 | 1,00,000 | 1,05,000 |
| **Transactions 3** | 1,20,000 | 1,10,000 | 1,15,000 |
| **Transactions 4** | 55,000 | 50000 | 52,500 |
| **Transactions 5** | 65,000 | 60000 | 62,500 |
| **Transactions 6** | 45,000 | 40,000 | 42,500 |
| **Transactions 7** | 45,000 | 41,000 | 43,000 |
| **Transactions 8** | 35,000 | 30,000 | 32,500 |

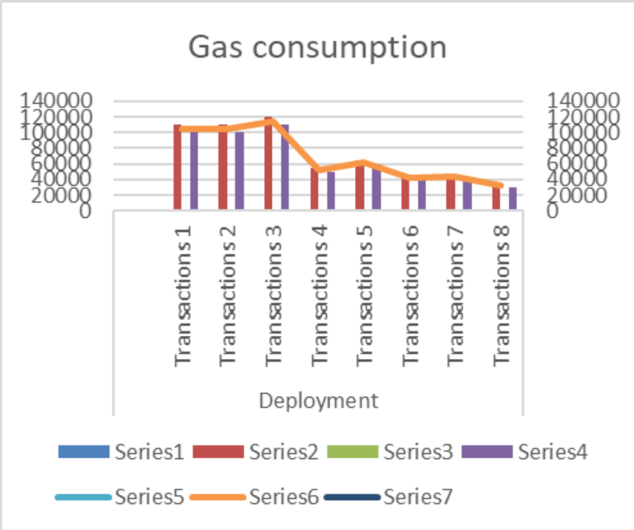


**Fig.2.** Comparison of Average Gas Cost.in each transaction

**Table 2.** comparison of blockchain-based frameworks

| s.no | Framework comparisons | | |
|---|---|---|---|
| | *Smart contract frameworks* | *Gas consumed* | *Cost in ether* |
| 1 | Ethereum Truffle | 6782836 | 0.008634 |
| 2 | Ethereum Hardhat | 3876764 | 0.087984 |
| 3 | Ethereum Open IPS | 4162277 | 0.097536 |
| 4 | Truffle AND Web.js | 2635428 | 0.002323 |
| 5 | Ethereum Hardhat | 4587382 | 0.004566 |
| 6 | Ethereum Truffle (proposed system) | 2343546 | 0.003426 |

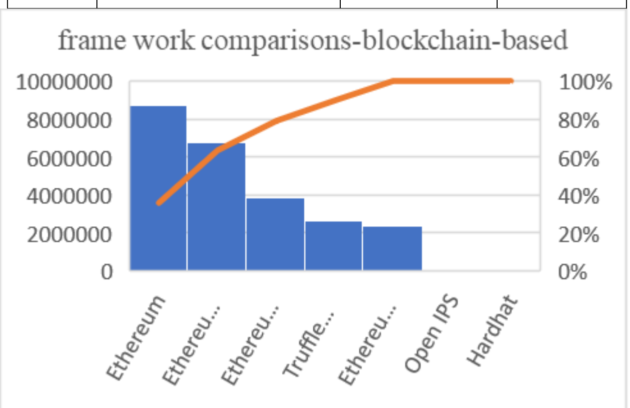


**Fig. 2.** Magnetization as a function of applied field. Note that "Fig." is abbreviated. There is a period after the figure number, followed by two spaces. It is good practice to explain the significance of the figure in the caption.

(i) Gas Consumption Metrics

The gas consumption of deploying and interacting with smart contracts, the following metrics will be calculated

Gas =∑_(i=1)^N 〖(Gas Units Used i*Gas price)〗 (6 )

Where,

Gas Units Used I is the, from Fig 1 and Fig2 gas used for the ith deployment.Gas price is the gas price at time of the ith deploymentN is the total number of deployment.

(ii) Gas cost Transactions

Gas =∑_(j=1)^N 〖(Gas Units Used j*Gas price)〗 (7)

Gas Units calculated for above in the jth transactions respectively.

(iii)Average Gas Usage For Deployment

Average Gas Usage =∑_(i=1)^N 〖(Gas Units Used i)/N〗 (8)

Where Gas Units Used I is the gas consumed for the ith deployment.

## 7.Conclusion

The estimation of gas for smart contract activities is a critical function of development frameworks. Within Hardhat, ethers.js is used for gas estimate using contract instances. Using the approach for predicting petrol and the associated ether consumption is part of this procedure. Similarly, gas estimate is possible with Truffle by utilizing gas invocation and contract.


## Acknowledgements

This research was supported . We thank our colleagues from Saveetha School of Engineering who provided insight and expertise that greatly assisted the research, although they may not agree with all of the interpretations of this paper. We thank [Name Surname, title] for assistance with [particular technique, methodology], and [Name Surname, position, institution name] for comments that greatly improved the manuscript.


## Author contributions

**Shakila Munuswamy:** Conceptualization, Methodology, Software, Field study **Rama Arunmozhi :** Data curation, Writing-Original draft preparation, Software, Validation., Field study **NP.Thilagavathiame:** Visualization, Investigation, Writing-Reviewing and Editing. **M.Prakash:** Investigation, Writing-Reviewing and Editing. **L.Anitha:** Data curation, Writing-Original draft preparation, Software, Validation

## Conflicts of interest

The authors declare no conflicts of interest.

**References**


[1] Alammary, A., Alhazmi, S., Almasri, M., & Gillani, S. (2019). Blockchain-based applications in education: A systematic review.

[2] (2023). Hands-up-go: Development of gas efficient Blockchain event DApp. doi: 10.1109/icoin56518.2023.10049034

[3] BitDegree. (2018). Bitdegree whitepaper. Retrieved from https://www.bitdegree.org/ bitdegree-vision.pdf (Accessed 29-09-18) Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[4] Organ Donation Facts and Info: Organ Transplants. Accessed: Apr. 18, 2021. [Online]. Available: https://my.clevelandclinic.org/health/ articles/11750-organ-donation-and-transplantation

[5] Organ Procurement and Transplantation Network. Accessed: Apr. 18, 2021. [Online]. Available: https://optn.transplant.hrsa.gov/ resources/ethics/ethical-principles-in-the-allocation-of-humanorgans

[6] S. N. Khan, "Blockchain smart contracts: Applications, challenges, and future Blockchain smart contracts: Applications, challenges, and future trends," no. April, 2021

[7] 8. E. Androulaki et al., "Hyperledger Fabric : A Distributed Operating System for Permissioned Blockchains."

[8] S. Steinhorst, "SmaCoNat : Smart Contracts in Natural Language," doi: 10.1109/FDL.2018.8524068.

[9] G Subhashini, M Karthi, A Pandi, S Gomathi, J Sivapriya, "Security System to Analyze, Recognize and Alert in Real Time using AI-Models" Second International Conference on Electronics and Renewable Systems (ICEARS), Tuticorin, India, 2023, pp. 754-760, doi: 10.1109/ICEARS56392.2023.10085421

[10] W. Zou et al., "Smart Contract Development : Challenges and Opportunities," no. March, pp. 1–20, 2018.

[11] G. Canfora, A. Di Sorbo, S. Laudanna, A. Vacca, and C. A. Visaggio, "Profiling Gas Leaks in Solidity Smart Contracts," 2020.

[12] T. Chen et al., "GasChecker : Scalable Analysis for Discovering Gas-Inefficient Smart Contracts," vol. 14, no. 8, 2020, doi: 10.1109/TETC.2020.2979019.

[13] T. Chen, X. Li, X. Luo, and X. Zhang, "Underoptimized smart contracts devour your money," SANER 2017 - 24th IEEE Int. Conf. Softw. Anal. Evol. Reengineering, pp. 442–446, 2017, doi: 10.1109/SANER.2017.7884650. 15. S. Bouraga, "An Evaluation of G

[14] Y. D. Yaga and D. Yaga, Blockchain Technology Overview Blockchain Technology Overview.

[15] S. Nakamoto, "Bitcoin : A Peer-to-Peer Electronic Cash System," pp. 1–9.

[16] Harshith Royal G,S Gomathi, D Sungeetha, "Enhancing the Accuracy of Block chain Based Agriculture Product Supply Chain Management System Using Random Forest Algorithm and Decision Tree Algorithm," International Conference on Intelligent Technologies for Sustainable Electric and Communications Systems (iTech SECOM), Coimbatore, India, 2023, pp. 163-166, doi: 10.1109/iTechSECOM59882.2023.10435088.

[17] 2nd ed., vol. 3, J. Peters, Ed. New York, NY, USA: McGraw-Hill, 1964, pp. 15–64.

[18] W.-K. Chen, Linear Networks and Systems. Belmont, CA, USA: Wadsworth, 1993, pp. 123–135.

[19] J. U. Duncombe, "Infrared navigation—Part I: An assessment of feasibility," *IEEE Trans. Electron Devices*, vol. ED-11, no. 1, pp. 34–39, Jan. 1959, 10.1109/TED.2016.2628402.

[20] E. P. Wigner, "Theory of traveling-wave optical laser," *Phys. Rev.*, vol. 134, pp. A635–A646, Dec. 1965.

[21] E. H. Miller, "A note on reflector arrays," *IEEE Trans. Antennas Propagat.*, to be published.

[22] E. E. Reber, R. L. Michell, and C. J. Carter, "Oxygen absorption in the earth's atmosphere," Aerospace Corp., Los Angeles, CA, USA, Tech. Rep. TR-0200 (4230-46)-3, Nov. 1988.

[23] J. H. Davis and J. R. Cogdell, "Calibration program for the 16-foot antenna," Elect. Eng. Res. Lab., Univ. Texas, Austin, TX, USA, Tech. Memo. NGL-006-69-3, Nov. 15, 1987.

[24] *Transmission Systems for Communications*, 3rd ed., Western Electric Co., Winston-Salem, NC, USA, 1985, pp. 44–60.

[25] *Motorola Semiconductor Data Manual*, Motorola Semiconductor Products Inc., Phoenix, AZ, USA, 1989.

[26] G. O. Young, "Synthetic structure of industrial plastics," in Plastics, vol. 3, Polymers of Hexadromicon, J. Peters, Ed., 2nd ed. New York, NY, USA: McGraw-Hill, 1964, pp. 15-64. [Online]. Available: http://www.bookref.com.

[27] *The Founders' Constitution*, Philip B. Kurland and Ralph Lerner, eds., Chicago, IL, USA: Univ. Chicago Press, 1987. [Online]. Available: http://press-pubs.uchicago.edu/founders/

[28] The Terahertz Wave eBook. ZOmega Terahertz Corp., 2014. [Online]. Available: http://dl.z-thz.com/eBook/zomega_ebook_pdf_1206_sr.pdf. Accessed on: May 19, 2014.

[29] Philip B. Kurland and Ralph Lerner, eds., *The Founders' Constitution.* Chicago, IL, USA: Univ. of Chicago Press, 1987, Accessed on: Feb. 28, 2010, [Online] Available: http://press-pubs.uchicago.edu/founders/

[30] J. S. Turner, "New directions in communications," *IEEE J. Sel. Areas Commun*., vol. 13, no. 1, pp. 11-23, Jan. 1995.

[31] W. P. Risk, G. S. Kino, and H. J. Shaw, "Fiber-optic frequency shifter using a surface acoustic wave incident at an oblique angle," *Opt. Lett.*, vol. 11, no. 2, pp. 115–117, Feb. 1986.

[32] P. Kopyt *et al., "*Electric properties of graphene-based conductive layers from DC up to terahertz range," *IEEE THz Sci. Technol.*, to be published. DOI: 10.1109/TTHZ.2016.2544142.

[33] PROCESS Corporation, Boston, MA, USA. Intranets: Internet technologies deployed behind the firewall for corporate productivity. Presented at INET96 Annual Meeting. [Online]. Available: http://home.process.com/Intranets/wp2.htp

[34] R. J. Hijmans and J. van Etten, "Raster: Geographic analysis and modeling with raster data," R Package Version 2.0-12, Jan. 12, 2012. [Online]. Available: http://CRAN.R-project.org/package=raster

[35] Teralyzer. Lytera UG, Kirchhain, Germany [Online]. Available: http://www.lytera.de/Terahertz_THz_Spectroscopy.php?id=home, Accessed on: Jun. 5, 2014

[36] U.S. House. 102nd Congress, 1st Session. (1991, Jan. 11). *H. Con. Res. 1, Sense of the Congress on Approval of Military Action*. [Online]. Available: LEXIS Library: GENFED File: BILLS

[37] Musical toothbrush with mirror, by L.M.R. Brooks. (1992, May 19). Patent D 326 189 [Online]. Available: NEXIS Library: LEXPAT File: DES

[38] D. B. Payne and J. R. Stern, "Wavelength-switched pas- sively coupled single-mode optical network," in *Proc. IOOC-ECOC,* Boston, MA, USA, 1985, pp. 585–590.

[39] D. Ebehard and E. Voges, "Digital single sideband detection for interferometric sensors," presented at the *2nd Int. Conf. Optical Fiber Sensors,* Stuttgart, Germany, Jan. 2-5, 1984.

[40] G. Brandli and M. Dick, "Alternating current fed power supply," U.S. Patent 4 084 217, Nov. 4, 1978.

[41] J. O. Williams, "Narrow-band analyzer," Ph.D. dissertation, Dept. Elect. Eng., Harvard Univ., Cambridge, MA, USA, 1993.

[42] N. Kawasaki, "Parametric study of thermal and chemical nonequilibrium nozzle flow," M.S. thesis, Dept. Electron. Eng., Osaka Univ., Osaka, Japan, 1993.